

AMENDMENTS TO THE CLAIMS

Please amend claims 23, 43, and 56, as follow. A complete listing of the currently pending claims is provided below.

Claims 1-22 (Cancelled)

23. (Currently Amended) A method of handling document operation requests, the method comprising:

receiving a document operation request, the document operation requiring one or more collection elements of the document to be in a computer-readable memory;
identifying at least one of the one or more required collection elements for processing;
determining whether the at least one identified collection element is within a collection partition in the computer-readable memory;
loading the at least one identified collection element into the computer-readable memory based on a result from the act of determining, wherein the at least one identified collection element is loaded into the computer-readable memory when the at least one identified collection element is not within a collection partition in the computer-readable memory; and
executing the document operation.

24. (Previously Presented) The method of claim 23, wherein a collection partition does not exceed a threshold size.

25. (Previously Presented) The method of claim 24, wherein the threshold size is a factor of the memory size.

26. (Previously Presented) The method of claim 24, wherein the threshold size is user defined.

27. (Previously Presented) The method of claim 23, wherein the memory is fixed in size.
28. (Previously Presented) The method of claim 23, wherein a collection partition is a unit in which data is written to or read from a data storage device.
29. (Previously Presented) The method of claim 23, wherein a collection partition comprises collection elements from one collection.
30. (Previously Presented) The method of claim 23, wherein a collection partition is a disjoint subset of a collection in the document.
31. (Previously Presented) The method of claim 23, wherein loading the at least one identified collection element into memory comprises:
loading the at least one identified collection element into a new collection partition in the memory when the at least one identified collection element is not within a collection partition in the memory.
32. (Previously Presented) The method of claim 23, further comprising:
determining whether the memory is full;
selecting one or more collection partitions in the memory for removal when the memory is full;
propagating one or more changes in each of the one or more selected collection partitions to one or more data storage devices storing one or more collection elements in the selected collection partition;
removing the one or more selected collection partitions from the memory; and
updating one or more collection partitions that remain in the memory.
33. (Previously Presented) The method of claim 32, wherein the memory is full when the memory is above a threshold.

34. (Previously Presented) The method of claim 33, wherein the threshold is user defined.
35. (Previously Presented) The method of claim 32, wherein selecting one or more collection partitions in the memory for removal comprises:

selecting one or more least recently used collection partitions in the memory for removal when the memory is full.

36. (Previously Presented) The method of claim 32, wherein updating one or more collection partitions comprises:

updating metadata corresponding to the one or more collection partitions that remain in the memory.

37. (Previously Presented) The method of claim 32, wherein the one or more selected collection partitions do not contain any of the one or more required collection elements.

38. (Previously Presented) The method of claim 23, wherein the document operation is a read, update, delete, insert, or create operation.

39. (Previously Presented) The method of claim 23, wherein the one or more required collection elements are part of one collection.

40. (Previously Presented) The method of claim 23, wherein determining whether the at least one identified collection element is within a collection partition in the memory comprises:

determining whether a collection partition corresponding to the at least one identified collection element has already been loaded into memory.

41. (Previously Presented) The method of claim 40, wherein loading the at least one identified collection element into memory comprises:

loading the corresponding collection partition into memory when the corresponding collection partition has not already been loaded into memory.

42. (Previously Presented) The method of claim 23, wherein the document is an XML document.

43. (Currently Amended) A computer program product that includes a computer readable medium, the computer readable medium being a non-volatile medium or a volatile medium, the computer readable medium comprising a plurality of instructions which, when executed by a processor, causes the processor to execute a process for handling document operation requests, the process comprising:

receiving a document operation request, the document operation requiring one or more collection elements of the document to be in memory;

identifying at least one of the one or more required collection elements for processing;

determining whether the at least one identified collection element is within a collection partition in the memory;

loading the at least one identified collection element into memory based on a result from the act of determining, wherein the at least one identified collection element is loaded into the memory when the at least one identified collection element is not within a collection partition in the memory; and

executing the document operation.

44. (Previously Presented) The computer program product of claim 43, wherein a collection partition does not exceed a user-defined threshold size.

45. (Previously Presented) The computer program product of claim 43, wherein the memory is fixed in size.

46. (Previously Presented) The computer program product of claim 43, wherein a collection partition is a unit in which data is written to or read from a data storage device.

47. (Previously Presented) The computer program product of claim 43, wherein a collection partition is a disjoint subset of a collection in the document.

48. (Previously Presented) The computer program product of claim 43, wherein loading the at least one identified collection element into memory comprises:

loading the at least one identified collection element into a new collection partition in the memory when the at least one identified collection element is not within a collection partition in the memory.

49. (Previously Presented) The computer program product of claim 43, wherein the process further comprises:

determining whether the memory is full;

selecting one or more collection partitions in the memory for removal when the memory is full;

propagating one or more changes in each of the one or more selected collection partitions to one or more data storage devices storing one or more collection elements in the selected collection partition;

removing the one or more selected collection partitions from the memory; and

updating one or more collection partitions that remain in the memory.

50. (Previously Presented) The computer program product of claim 49, wherein the memory is full when the memory is above a user-defined threshold.

51. (Previously Presented) The computer program product of claim 49, wherein selecting one or more collection partitions in the memory for removal comprises:

selecting one or more least recently used collection partitions in the memory for removal when the memory is full.

52. (Previously Presented) The computer program product of claim 49, wherein updating one or more collection partitions comprises:

updating metadata corresponding to the one or more collection partitions that remain in the memory.

53. (Previously Presented) The computer program product of claim 43, wherein the document operation is a read, update, delete, insert, or create operation.

54. (Previously Presented) The computer program product of claim 43, wherein determining whether the at least one identified collection element is within a collection partition in the memory comprises:

determining whether a collection partition corresponding to the at least one identified collection element has already been loaded into memory.

55. (Previously Presented) The computer program product of claim 54, wherein loading the at least one identified collection element into memory comprises:

loading the corresponding collection partition into memory when the corresponding collection partition has not already been loaded into memory.

56. (Currently Amended) A system for handling document operation requests, the system comprising:

means for receiving a document operation request, the document operation requiring one or more collection elements of the document to be in memory;

means for identifying at least one of the one or more required collection elements for processing;

means for determining whether the at least one identified collection element is within a collection partition in the memory;

means for loading the at least one identified collection element into memory based on a result from the act of determining, wherein the means for loading loads the at least one identified collection element into the memory when the at least one identified collection element is not within a collection partition in the memory; and

means for executing the document operation.

57. (Previously Presented) The system of claim 56, wherein a collection partition does not exceed a user-defined threshold size.

58. (Previously Presented) The system of claim 56, wherein the memory is fixed in size.

59. (Previously Presented) The system of claim 56, wherein a collection partition is a unit in which data is written to or read from a data storage device.

60. (Previously Presented) The system of claim 56, wherein a collection partition is a disjoint subset of a collection in the document.

61. (Previously Presented) The system of claim 56, wherein means for loading the at least one identified collection element into memory comprises:

means for loading the at least one identified collection element into a new collection partition in the memory when the at least one identified collection element is not within a collection partition in the memory.

62. (Previously Presented) The system of claim 56, further comprising:

means for determining whether the memory is full;

means for selecting one or more collection partitions in the memory for removal when the memory is full;

means for propagating one or more changes in each of the one or more selected collection partitions to one or more data storage devices storing one or more collection elements in the selected collection partition;

means for removing the one or more selected collection partitions from the memory; and

means for updating one or more collection partitions that remain in the memory.

63. (Previously Presented) The system of claim 62, wherein the memory is full when the memory is above a user-defined threshold.

64. (Previously Presented) The system of claim 62, wherein means for selecting one or more collection partitions in the memory for removal comprises:

means for selecting one or more least recently used collection partitions in the memory for removal when the memory is full.

65. (Previously Presented) The system of claim 62, wherein means for updating one or more collection partitions comprises:

means for updating metadata corresponding to the one or more collection partitions that remain in the memory.

66. (Previously Presented) The system of claim 56, wherein the document operation is a read, update, delete, insert, or create operation.

67. (Previously Presented) The system of claim 56, wherein means for determining whether the at least one identified collection element is within a collection partition in the memory comprises:

means for determining whether a collection partition corresponding to the at least one identified collection element has already been loaded into memory.

68. (Previously Presented) The system of claim 67, wherein means for loading the at least one identified collection element into memory comprises:

means for loading the corresponding collection partition into memory when the corresponding collection partition has not already been loaded into memory.